

A Partial EM Algorithm for Clustering White Breads

Ryan P. Browne*, Paul D. McNicholas*, Christopher J. Findlay†

Abstract

The design of new products for consumer markets has undergone a major transformation over the last 50 years. Traditionally, inventors would create a new product that they thought might address a perceived need of consumers. Such products tended to be developed to meet the inventors own perception and not necessarily that of consumers. The social consequence of a top-down approach to product development has been a large failure rate in new product introduction. By surveying potential customers, a refined target is created that guides developers and reduces the failure rate. Today, however, the proliferation of products and the emergence of consumer choice has resulted in the identification of segments within the market. Understanding your target market typically involves conducting a product category assessment, where 12 to 30 commercial products are tested with consumers to create a preference map. Every consumer gets to test every product in a complete-block design; however, many classes of products do not lend themselves to such approaches because only a few samples can be evaluated before ‘fatigue’ sets in. We consider an analysis of incomplete balanced-incomplete-block data on 12 different types of white bread. A latent Gaussian mixture model is used for this analysis, with a partial expectation-maximization (PEM) algorithm developed for parameter estimation. This PEM algorithm circumvents the need for a traditional E-step, by performing a partial E-step that reduces the Kullback-Leibler divergence between the conditional distribution of the missing data and the distribution of the missing data given the observed data. The results of the white bread analysis are discussed and some mathematical details are given in an appendix.

Keywords: Balanced-incomplete-block; mixture models; progressive EM; sensometrics; white bread.

1 Introduction

Consumer-driven product development of new consumer products and the improvement of existing products have become recognized as a best-practice in industry. Food researchers have become increasingly dependent on understanding consumer wants and desires to effectively design food products (Jaeger et al., 2003). To understand consumer behaviour, preference maps are built by assessing consumer liking of an appropriate range of commercial products within a category. From these liking data, a model may be built that describes the ideal product for the test population. However, most product categories will have more than a single ideal product, with two or more liking clusters revealed. Hedonic taste tasting is the most common practice used to measure consumer liking within a target population (Lawless and Heymann, 2010). In a complete-block design, every consumer gets to taste every product, but many product categories do not facilitate this sampling plan. When tasting wine, for example, a consumer can only evaluate three or four samples before ‘fatigue’ sets in, compromising the quality of the data collected. The fact that consumers tend to behave like experts puts into doubt the value of data obtained over multiple tasting sessions (Findlay, 2008). Therefore, a balanced-incomplete-block (BIB) design is used for high-fatigue products. The resulting data are sparse and tend to be heterogenous; therefore, we must identify sub-populations in an incomplete-data setting.

*Department of Mathematics & Statistics, University of Guelph, Guelph, Ontario, N1G 2W1, Canada. E-mail: {rbrowne,paul.mcnicholas}@uoguelph.ca.

†Compusense Inc., Guelph, Ontario, N1G 4S2, Canada. E-mail: cfindlay@compusense.com

In this paper, we consider an analysis of 12 white breads. The descriptive analysis of the breads provides a measure of the range of sensory properties found within the product category. It also gives us information regarding changes that may improve the sensory liking of a product for a specific consumer segment. Consumer research is conducted to measure liking on a nine-point hedonic scale anchored at dislike extremely (1) and like extremely (9), with the midpoint (5) indicating neither like nor dislike. By clustering consumers on similarity of liking profiles across the products, it is possible to determine the sensory attributes that contribute to like/dislike within each cluster. The calibrated descriptive analysis was performed by a trained panel using well-defined product attributes that have been rated for intensity on a scale from 0 to 100. The attributes are generated to provide a complete sensory description of the breads. Some attributes are found at low intensity, but are important in differentiating products. There are also attributes that are defined by a major attribute or group of attributes. For example, sourdough breads would score high in sourness and sour aroma and flavour. The products selected for this study encompass the commercial sliced white bread category. The range goes from the extremely popular sandwich breads that are fine-celled, spongy, and bland to a ciabatta-style Italian hearth bread. The breads differ in crust colour and roughness, texture of the crumb, and flavour, but all fall within the realm of sliced white bread. A total of 369 consumers evaluated six breads within a 12-present-6 BIB design.

One straightforward way to tackle such an incomplete-data problem is to impute the missing data prior to the analysis. However, this approach is not generally desirable for clustering problems because the imputed values will be partly based on data from other sub-populations. Herein, we develop a clustering approach for these data based on a finite Gaussian mixture model. A random variable \mathbf{X} follows a G -component finite Gaussian mixture model if its density can be written

$$f(\mathbf{x} | \boldsymbol{\vartheta}) = \sum_{g=1}^G \pi_g \phi(\mathbf{x} | \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g), \quad (1)$$

where $\pi_g > 0$, with $\sum_{g=1}^G \pi_g = 1$, are mixing proportions, $\phi(\mathbf{x} | \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$ is multivariate Gaussian density with mean $\boldsymbol{\mu}_g$ and covariance matrix $\boldsymbol{\Sigma}_g$, and $\boldsymbol{\vartheta} = (\pi_1, \dots, \pi_G, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_G, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_G)$. Finite mixture models have been used for clustering for at least fifty years (Wolfe, 1963) and such applications are commonly referred to as ‘model-based clustering’ (cf. Fraley and Raftery, 2002). One problem with applications of Gaussian mixture models is the number of free covariance parameters: $Gp(p+1)/2$. To overcome this, many authors have considered imposing constraints on decomposed component covariance matrices (e.g., Celeux and Govaert, 1995) and other have considered underlying latent factor models (e.g., Ghahramani and Hinton, 1997). We consider an underlying latent factor model herein (cf. Section 3) and because so many of the data are missing, we assume common component covariance. The result is a parsimonious Gaussian mixture model.

The expectation-maximization (EM) algorithm (Dempster et al., 1977) is the standard approach to parameter estimation for model-based clustering (cf. McLachlan and Basford, 1988). However, in our incomplete-block data we obtain only 6 liking scores from 12 products for each consumer; therefore, an EM algorithm would require $\binom{12}{6} = 924$ different 6×6 matrix inversions for each mixture component in each E-step. To circumvent this problem, we develop a ‘partial’ EM (PEM) algorithm that requires only a single 12×12 matrix inversion for each mixture component. We show that this PEM retains the monotonicity property and thus all of the convergence properties of the standard EM algorithm, but is much more computationally efficient than the standard EM algorithm for this particular problem.

The remainder of this paper is laid out as follows. In Section 2, we review the application of the EM algorithm for missing data. Then, our parsimonious Gaussian mixture model is presented and the PEM algorithm is developed (Section 3). We apply our method to the white bread data in Section 4, where we also compare our PEM algorithm to the standard EM algorithm (Section 4.3). The paper concludes with discussion and suggestions for future work (Section 5).

2 The EM Algorithm for Missing Data Problems

The EM algorithm is an iterative procedure for finding maximum likelihood estimates when data are incomplete. Therefore, EM algorithms are naturally suited for missing data problems. The EM algorithm consists of alternating between E- and M-steps until a convergence criterion is satisfied. In the E-step, the expected value of the complete-data (i.e., the observed plus missing data) is computed, and in the M-step, this quantity is maximized with respect to the parameters. Formally, the EM algorithm is a special case of an MM algorithm of the minorization-maximization variety (Hunter and Lange, 2000, 2004).

Suppose we observe p -dimensional $\mathbf{y}_1, \dots, \mathbf{y}_n$ such that each \mathbf{y}_i can be decomposed into an observed component, \mathbf{x}_i , of dimension m , a missing component, \mathbf{z}_i , of dimension $l = p - m$, and

$$\mathbf{Y}_i = \begin{bmatrix} \mathbf{X}_i \\ \mathbf{Z}_i \end{bmatrix} \sim \mathcal{N}\left(\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_x \\ \boldsymbol{\mu}_z \end{bmatrix}, \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{xx} & \boldsymbol{\Sigma}_{xz} \\ \boldsymbol{\Sigma}_{zx} & \boldsymbol{\Sigma}_{zz} \end{bmatrix}\right).$$

Then, the conditional distribution of the missing data given the observed is given by

$$\mathbf{Z}_i | \mathbf{X}_i = \mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\mu}_{z_i|x_i} := \boldsymbol{\mu}_z + \boldsymbol{\Sigma}_{zx} \boldsymbol{\Sigma}_{xx}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_x), \boldsymbol{\Sigma}_{z|x} := \boldsymbol{\Sigma}_{zz} - \boldsymbol{\Sigma}_{zx} \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\Sigma}_{xz}).$$

Now, define

$$\hat{\mathbf{z}}_i := \boldsymbol{\mu}_{z_i|x_i} = \boldsymbol{\mu}_z + \boldsymbol{\Sigma}_{zx} \boldsymbol{\Sigma}_{xx}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_x), \quad (2)$$

$$\hat{\mathbf{Y}}_i := \begin{bmatrix} \mathbf{0}_{l \times l} & \mathbf{0}_{l \times m} \\ \mathbf{0}_{l \times m} & \hat{\mathbf{Z}}_i \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{l \times l} & \mathbf{0}_{l \times m} \\ \mathbf{0}_{m \times l} & \boldsymbol{\Sigma}_{z|x} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{l \times l} & \mathbf{0}_{l \times m} \\ \mathbf{0}_{m \times l} & \boldsymbol{\Sigma}_{zz} - \boldsymbol{\Sigma}_{zx} \boldsymbol{\Sigma}_{xx}^{-1} \boldsymbol{\Sigma}_{xz} \end{bmatrix}, \quad (3)$$

where $\mathbf{0}_{l \times m}$ is an $l \times m$ matrix of zeros and $\hat{\mathbf{y}}_i = (\mathbf{x}_i, \hat{\mathbf{z}}_i)$. Using this notation, the EM updates for the mean and covariance can be written as

$$\hat{\boldsymbol{\mu}}^{(t+1)} = \bar{\mathbf{y}} = \frac{1}{n} \sum_{i=1}^n \hat{\mathbf{y}}_i \quad \text{and} \quad \hat{\boldsymbol{\Sigma}}^{(t+1)} = \mathbf{S} = \sum_{i=1}^n (\hat{\mathbf{y}}_i - \bar{\mathbf{y}}) (\hat{\mathbf{y}}_i - \bar{\mathbf{y}})' + \sum_{i=1}^n \hat{\mathbf{Y}}_i, \quad (4)$$

respectively. Because our white bread data have lots of missing observations, ‘standard’ E-steps are very computationally expensive. For example, each observation requires inversion of a 6×6 matrix; this amounts to $\binom{12}{6} = 920$ different matrix inversions at each iteration.

Next, consider approximate E-steps instead of full E-steps. All of these procedures will work with the inverse of $\boldsymbol{\Sigma}$ and its principal sub-matrices and vectors. We will assume that $\boldsymbol{\Sigma}^{-1}$ is known; this quantity is typically readily available because it is necessary to calculate the log-likelihood at each EM iteration. We denote

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{xx} & \boldsymbol{\Sigma}_{xz} \\ \boldsymbol{\Sigma}_{zx} & \boldsymbol{\Sigma}_{zz} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\Sigma}^{-1} = \boldsymbol{\Xi} = \begin{bmatrix} \boldsymbol{\Xi}_{xx} & \boldsymbol{\Xi}_{xz} \\ \boldsymbol{\Xi}_{zx} & \boldsymbol{\Xi}_{zz} \end{bmatrix}$$

because there exists a relationship between $\boldsymbol{\Xi}_{zz}$ and $\boldsymbol{\Sigma}_{z|x}$. This relationship exists because $\boldsymbol{\Sigma}_{z|x}$ is the Schur complement of the matrix $\boldsymbol{\Sigma}$ which has the property that

$$\boldsymbol{\Sigma}_{z|x} = \boldsymbol{\Xi}_{zz}^{-1} \quad \text{and, equivalently,} \quad \boldsymbol{\Sigma}_{z|x}^{-1} = \boldsymbol{\Xi}_{zz}.$$

In addition, there exists a relationship between the regression coefficients and $\boldsymbol{\Xi}$, which can be derived through block inversion of $\boldsymbol{\Sigma}$,

$$\boldsymbol{\Sigma}_{zx} \boldsymbol{\Sigma}_{xx}^{-1} = -\boldsymbol{\Xi}_{zz}^{-1} \boldsymbol{\Xi}_{zx}.$$

These relations can be useful if the dimension of \mathbf{z} is smaller than the dimension of \mathbf{x} . For example, if there is a single missing observation then using $\boldsymbol{\Sigma}$ requires a $p - 1$ matrix inversion, whereas if $\boldsymbol{\Xi}$ is known we only require an inversion of a 1×1 matrix.

For the extension $G > 1$ mixture components, we require the weighted versions of (4). The weight for observation i in component g is

$$w_{ig} = \frac{\pi_g \phi(\mathbf{x}_i | \boldsymbol{\mu}_{g,x}, \boldsymbol{\Sigma}_{g,xx})}{\sum_{k=1}^G \pi_k \phi(\mathbf{x}_i | \boldsymbol{\mu}_{k,x}, \boldsymbol{\Sigma}_{k,xx})}, \quad (5)$$

where $\hat{\pi}_g^{(t+1)} = n_g/n = (1/n) \sum_{i=1}^n w_{ig}$, $\hat{\boldsymbol{\mu}}_g^{(t+1)} = \bar{\mathbf{y}}_g = (1/n_g) \sum_{i=1}^n w_{ig} \hat{\mathbf{y}}_i$, and

$$\hat{\boldsymbol{\Sigma}}_g^{(t+1)} = \mathbf{S}_g = \frac{1}{n_g} \sum_{i=1}^n \left[w_{ig} (\hat{\mathbf{y}}_i - \bar{\mathbf{y}}) (\hat{\mathbf{y}}_i - \bar{\mathbf{y}})' + \hat{\mathbf{Y}}_i \right], \quad (6)$$

where $n_g = \sum_{i=1}^n w_{ig}$.

3 Methodology

3.1 Finite Mixture Models with Common Factors

As already mentioned (Section 1), it is common practice to introduce parsimony via constrained component covariance matrices. When dealing with sparse data, such as the white bread data, estimating the covariance matrix can be especially difficult. Therefore, we use a variant of the mixture of factor analyzers model (Ghahramani and Hinton, 1997; McLachlan and Peel, 2000) in which we constrain the component factor loading matrices to be equal across groups (cf. McNicholas and Murphy, 2008). The factor analysis model (Spearman, 1904; Bartlett, 1953) assumes that a p -dimensional random vector \mathbf{X}_i can be modelled using a q -dimensional vector of latent factors \mathbf{U}_i , where $q \ll p$. The model can be written $\mathbf{X}_i = \boldsymbol{\mu} + \boldsymbol{\Lambda} \mathbf{U}_i + \boldsymbol{\epsilon}$, where $\boldsymbol{\Lambda}$ is a $p \times q$ matrix of factor weights, the latent variables $\mathbf{U}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_q)$, and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Psi})$, where $\boldsymbol{\Psi}$ is a $p \times p$ diagonal matrix. Therefore, the marginal distribution of \mathbf{X}_i is $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Lambda} \boldsymbol{\Lambda}' + \boldsymbol{\Psi})$. It follows that the density for the mixture of factor analyzers model is that of Equation 1 with $\boldsymbol{\Sigma}_g = \boldsymbol{\Lambda}_g \boldsymbol{\Lambda}_g' + \boldsymbol{\Psi}_g$. The model we use for the analysis of the bread data assumes equal factor loading matrices across components, i.e., $\boldsymbol{\Sigma}_g = \boldsymbol{\Lambda} \boldsymbol{\Lambda}' + \boldsymbol{\Psi}_g$, and so the EM algorithm updates are given by

$$\text{vec} \left(\hat{\boldsymbol{\Lambda}}^{(\text{new})} \right) = \left[\sum_{g=1}^G n_g \hat{\boldsymbol{\Psi}}_g^{-1} \otimes \hat{\boldsymbol{\Theta}}_g \right]^{-1} \text{vec} \left(\sum_{g=1}^G n_g \hat{\boldsymbol{\Psi}}_g^{-1} \mathbf{S}_g \boldsymbol{\beta}_g' \right), \quad (7)$$

$$\hat{\boldsymbol{\Psi}}_g^{(\text{new})} = \text{diag} \left\{ \mathbf{S}_g - 2 \hat{\boldsymbol{\Lambda}}^{(\text{new})} \hat{\boldsymbol{\beta}}_g \mathbf{S}_g + \hat{\boldsymbol{\Lambda}}^{(\text{new})} \hat{\boldsymbol{\Theta}}_g \left(\hat{\boldsymbol{\Lambda}}^{(\text{new})} \right)' \right\}. \quad (8)$$

3.2 PEM Algorithm

We follow Neal and Hinton (1998) and store the sufficient statistics $(\hat{\mathbf{z}}_1, \dots, \hat{\mathbf{z}}_n)$ and $(\hat{\mathbf{Z}}_1, \dots, \hat{\mathbf{Z}}_n)$. However, instead of doing a complete update of a partial set of the sufficient statistics as suggested by Neal and Hinton (1998), we perform a partial E-step at each iteration. These partial E-steps can be shown to reduce the Kullback-Leibler (KL) divergence at every step, which ensures that the monotonicity of the EM algorithm is preserved. From Neal and Hinton (1998), the EM algorithm can be viewed as minimizing the function

$$\sum_{i=1}^n F(\tilde{P}_i, \theta) = - \sum_{i=1}^n D(\tilde{P}_i || P_{i,\theta}) + \sum_{i=1}^n l(\mathbf{x}_i | \theta),$$

where $D(\tilde{P}_i || P_{i,\theta})$ is the KL divergence between the distribution of the missing data \tilde{P}_i and the conditional distribution of the missing data given the observed data, $P_\theta = P(\mathbf{Z}_i | \mathbf{x}_i, \theta)$. A ‘standard’ E-step sets \tilde{P}_i

to $P(\mathbf{Z}_i|\mathbf{x}_i, \theta_t)$, for all i , at each iteration t . Neal and Hinton (1998) suggest a partial or sparse E-step, where a subset of \tilde{P}_i is updated to $P(\mathbf{Z}_i|\mathbf{x}_i, \theta_t)$ at each EM iteration. The algorithm we describe in the next two sections partially updates each \tilde{P}_i towards $P(\mathbf{Z}_i|\mathbf{x}_i, \theta)$ such that the KL divergence is reduced but not minimized. For the multivariate Gaussian distribution and a particular i , the EM algorithm can be viewed as minimizing

$$F(N_{\mathbf{z}}, \mathbf{x}_i, \theta) = -D_{\text{KL}}(N_{\mathbf{z}}||N_{\mathbf{z}, \mathbf{x}_i}) + l(\mathbf{x}_i|\theta),$$

with respect to $N_{\mathbf{z}}$, the distribution of the latent or missing variables, and the parameter set θ .

The KL divergence between the missing data distribution with mean $\hat{\mathbf{z}}_i$ and variance $\hat{\mathbf{Z}}_i$ and the conditional distribution of the missing data given the observed data \mathbf{x}_i is

$$D_{\text{KL}}(N_{\mathbf{z}}||N_{\mathbf{z}, \mathbf{x}_i}) = \frac{1}{2} \left[\text{tr} \left\{ \mathbf{\Sigma}_{\mathbf{z}, \mathbf{x}_i}^{-1} \hat{\mathbf{Z}}_i \right\} + (\hat{\mathbf{z}}_i - \boldsymbol{\mu}_{\mathbf{z}, \mathbf{x}_i})' \mathbf{\Sigma}_{\mathbf{z}, \mathbf{x}_i}^{-1} (\hat{\mathbf{z}}_i - \boldsymbol{\mu}_{\mathbf{z}, \mathbf{x}_i}) - \ln \left(\frac{|\hat{\mathbf{Z}}_i|}{|\mathbf{\Sigma}_{\mathbf{z}, \mathbf{x}_i}|} \right) \right]. \quad (9)$$

From Equation 9, we can see that if we set $\hat{\mathbf{z}}_i$ to the conditional mean and $\hat{\mathbf{Z}}_i$ to the conditional covariance matrix, then the KL divergence is minimized. However, for our data this involves inverting $\binom{12}{6} = 924$ different 6×6 matrices at each iteration. Finding the minimum distribution for the missing data in each row is computationally expensive, so we will instead iteratively minimize the KL divergence on simpler computations.

3.3 Notation

Hereafter, the following notation will be used. Let $\mathbf{\Sigma}_j$ be the principal sub-matrix of $\mathbf{\Sigma}$, obtained by deleting column j and row j . Let σ_j and ξ_j be the j th diagonal elements of $\mathbf{\Sigma}$ and $\mathbf{\Xi}$, respectively. Let $\boldsymbol{\sigma}_j$ and $\boldsymbol{\xi}_j$ be the j th rows of $\mathbf{\Sigma}$ and $\mathbf{\Xi}$, respectively, with the j th element deleted. For example, if $j = 1$, then

$$\mathbf{\Sigma} = \begin{bmatrix} \sigma_1 & \boldsymbol{\sigma}'_1 \\ \boldsymbol{\sigma}_1 & \mathbf{\Sigma}_1 \end{bmatrix} \quad \text{and} \quad \mathbf{\Sigma}^{-1} = \mathbf{\Xi} = \begin{bmatrix} \xi_1 & \boldsymbol{\xi}'_1 \\ \boldsymbol{\xi}_1 & \mathbf{\Xi}_1 \end{bmatrix}. \quad (10)$$

3.4 Minimizing the KL Divergence With Respect to $\hat{\mathbf{z}}_i$

To minimize the KL divergence with respect to $\hat{\mathbf{z}}_i$, we set it to $\boldsymbol{\mu}_{\mathbf{z}, \mathbf{x}_i}$, which is given in (2). However, $\boldsymbol{\mu}_{\mathbf{z}, \mathbf{x}_i}$ depends on a matrix inversion that depends on i . We now develop an updating equation that reduces the number of matrix inversions. The KL divergence depends on $\hat{\mathbf{z}}_i$ through one term in equation (9) and

$$\arg\max_{\hat{\mathbf{z}}_i} D_{\text{KL}}(N_{\mathbf{z}}||N_{\mathbf{z}, \mathbf{x}_i}) = \arg\max_{\hat{\mathbf{z}}_i} (\hat{\mathbf{y}}_i - \boldsymbol{\mu})' \mathbf{\Sigma}^{-1} (\hat{\mathbf{y}}_i - \boldsymbol{\mu})$$

because the last term in

$$(\hat{\mathbf{y}}_i - \boldsymbol{\mu})' \mathbf{\Sigma}^{-1} (\hat{\mathbf{y}}_i - \boldsymbol{\mu}) = (\hat{\mathbf{z}}_i - \boldsymbol{\mu}_{\mathbf{z}, \mathbf{x}_i})' \mathbf{\Sigma}_{\mathbf{z}, \mathbf{x}_i}^{-1} (\hat{\mathbf{z}}_i - \boldsymbol{\mu}_{\mathbf{z}, \mathbf{x}_i}) + (\mathbf{x}_i - \boldsymbol{\mu}_x)' \mathbf{\Sigma}_{xx}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_x) \quad (11)$$

is fixed and, moreover, does not depend on \mathbf{z}_i (cf. Appendix A.1).

We now consider a conditional minimization, by each element in $\mathbf{y}_i = (y_{i1}, \dots, y_{in})$, of

$$\arg\max_{\hat{\mathbf{z}}_i} (\hat{\mathbf{y}}_i - \boldsymbol{\mu})' \mathbf{\Sigma}^{-1} (\hat{\mathbf{y}}_i - \boldsymbol{\mu}),$$

which is given by

$$y_{ij}^{(t)} = \begin{cases} \mu_j + \boldsymbol{\sigma}_j \mathbf{\Sigma}_j^{-1} (y_{-j}^{(t)} - \boldsymbol{\mu}_{-k}) & \text{if } j \text{ corresponds to an element in } \hat{\mathbf{z}}_i, \\ y_{ij} & \text{if } j \text{ corresponds to an element in } \mathbf{x}_i, \end{cases}$$

where μ_k is the k th element of $\boldsymbol{\mu}$. This procedure requires inversion of the $\boldsymbol{\Sigma}_j$, i.e., the j th principal sub-matrix; which involves a $p - 1$ matrix inversion. However, under the assumption that $\boldsymbol{\Sigma}^{-1} = \boldsymbol{\Xi}$ is known, the updates can be simplified to

$$y_{ij}^{(t)} = \begin{cases} \mu_j - \frac{1}{\xi_j} \boldsymbol{\xi}_j \left(y_{-j}^{(t)} - \boldsymbol{\mu}_{-j} \right) & \text{if } j \text{ is corresponds to an element in } \hat{\mathbf{z}}_i, \\ y_{ij} & \text{if } j \text{ is corresponds to an element in } \mathbf{x}_i. \end{cases}$$

This set of updates requires the inversion of a 1×1 matrix, which is trivial. These updates are guaranteed to converge to the global minimum because the objective function is convex. The advantage of this conditional minimization of the KL divergence is that the update is computationally simple. If we let the $n \times p$ matrix $\hat{\mathbf{A}} = (\hat{\mathbf{y}}'_1, \dots, \hat{\mathbf{y}}'_n)$, then our partial E-steps update $\hat{\mathbf{A}}$ by column instead of ‘standard’ E-steps that update $\hat{\mathbf{A}}$ by row.

3.5 Minimizing the KL Divergence With Respect to $\hat{\mathbf{Z}}_i$

Now, minimizing the KL divergence with respect to $\hat{\mathbf{Z}}_i$ we have

$$\operatorname{argmax}_{\hat{\mathbf{Z}}_i} D_{\text{KL}}(N_z || N_{z.x_i}) = \operatorname{argmax}_{\hat{\mathbf{Z}}_i} \operatorname{tr} \left\{ \boldsymbol{\Sigma}_{z.x}^{-1} \hat{\mathbf{Z}}_i \right\} - \ln \left(\frac{|\hat{\mathbf{Z}}_i|}{|\boldsymbol{\Sigma}_{z.x}|} \right).$$

The log-determinant and the trace function are convex with respect to the positive definite matrices (Magnus and Neudecker, 1998). Now, consider the function

$$\gamma(\mathbf{Z}_i) = \operatorname{tr} \left[\left(\boldsymbol{\Sigma} - \hat{\mathbf{Y}}_i \right) \boldsymbol{\Sigma}^{-1} \left(\boldsymbol{\Sigma} - \hat{\mathbf{Y}}_i \right) \right], \quad (12)$$

which is also convex, and these functions have the property

$$\operatorname{argmax}_{\hat{\mathbf{Z}}_i} D_{\text{KL}}(N_z || N_{z.x_i}) = \operatorname{argmax}_{\hat{\mathbf{Z}}_i} \operatorname{tr} \left[\left(\boldsymbol{\Sigma} - \hat{\mathbf{Y}}_i \right) \boldsymbol{\Sigma}^{-1} \left(\boldsymbol{\Sigma} - \hat{\mathbf{Y}}_i \right) \right]. \quad (13)$$

Both objective functions are minimized by the Schur complements (c.f. Appendix A.2 for the function γ). In addition, if one objective function is reduced then so is the other because both functions are convex and have the same (global) minimum. Therefore, if the function γ is reduced at every iteration, then the KL divergence is reduced at every iteration and so the algorithm has the monotonicity property.

Conditional updates for the function γ are derived in Appendix A.2. We update $\hat{\mathbf{Y}}_i$ by column and row, while holding the associated principal sub-matrix fixed. Therefore, if we denote the j th row of $\hat{\mathbf{Y}}_i$ by $\hat{\mathbf{Y}}_{i,j}$ and the $(p - 1) \times p$ matrix obtained from removing the j th row as $\hat{\mathbf{Y}}_{i,-j}$, then the updates can be written as

$$\hat{\mathbf{Y}}_{i,j}^{(t+1)} = \begin{cases} \boldsymbol{\sigma}_j + \left(\boldsymbol{\sigma}_j - \hat{\mathbf{Y}}_{i,j}^{(t)} \right)' \boldsymbol{\Sigma}_j^{-1} \left(\boldsymbol{\Sigma}_{-j} - \hat{\mathbf{Y}}_{i,-j}^{(t)} \right) & \text{if } j \text{ is associated with } \hat{\mathbf{z}}_i, \\ \mathbf{0}_{1 \times p} & \text{if } j \text{ is associated with } \mathbf{x}_i, \end{cases}$$

and then we set the j th column of $\hat{\mathbf{Y}}_i^{(t+1)}$ equal to the j th row.

We can avoid the matrix inversion of the principal sub-matrix $\boldsymbol{\Sigma}_j$ by again exploiting the properties of the inverse of $\boldsymbol{\Xi}$. Specifically, if $\boldsymbol{\xi}'_i \boldsymbol{\sigma}_i \neq 1$, then

$$\boldsymbol{\Sigma}_j^{-1} = \left[\mathbf{I}_{p-1} + \frac{1}{1 - \boldsymbol{\xi}'_j \boldsymbol{\sigma}_j} \boldsymbol{\xi}_j \boldsymbol{\sigma}'_j \right] \boldsymbol{\Xi}_j,$$

where \mathbf{I}_{p-1} is the $(p - 1) \times (p - 1)$ identity matrix.

3.6 Evaluating Weights and the Likelihood Function

The likelihood depends only on the observed data

$$f(\mathbf{x}_i) = \frac{1}{(2\pi)^{p/2} |\boldsymbol{\Sigma}_{xx}|^{1/2}} \exp \left\{ -\frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu}_x)' \boldsymbol{\Sigma}_{xx}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_x) \right\}. \quad (14)$$

However, from Equation (11) we have

$$(\hat{\mathbf{y}}_i - \boldsymbol{\mu})' \boldsymbol{\Sigma}^{-1} (\hat{\mathbf{y}}_i - \boldsymbol{\mu}) = (\hat{\mathbf{z}}_i - \boldsymbol{\mu}_{z,x})' \boldsymbol{\Sigma}_{z,x}^{-1} (\hat{\mathbf{z}}_i - \boldsymbol{\mu}_{z,x}) + (\mathbf{x}_i - \boldsymbol{\mu}_x)' \boldsymbol{\Sigma}_{xx}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_x). \quad (15)$$

Therefore,

$$(\mathbf{x}_i - \boldsymbol{\mu}_x)' \boldsymbol{\Sigma}_{xx}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_x) \leq (\hat{\mathbf{y}}_i - \boldsymbol{\mu})' \boldsymbol{\Sigma}^{-1} (\hat{\mathbf{y}}_i - \boldsymbol{\mu}) = (\hat{\mathbf{y}}_i - \boldsymbol{\mu})' \boldsymbol{\Xi} (\hat{\mathbf{y}}_i - \boldsymbol{\mu}) \quad (16)$$

and we have equality when $\hat{\mathbf{z}}_i = \boldsymbol{\mu}_{z,x}$. Our algorithm for $\hat{\mathbf{z}}_i$ will converge to $\boldsymbol{\mu}_{z,x}$, so if we use this approximation for the likelihood and weights calculations they will also converge to the true quantities.

To calculate $|\boldsymbol{\Sigma}_{xx}|$, we use the relationship between Schur complements and the determinant, and the relationship between the inverse matrix and the Schur complement;

$$\ln |\boldsymbol{\Sigma}| = \ln |\boldsymbol{\Sigma}_{xx}| + \ln |\boldsymbol{\Sigma}_{z,x}| = \ln |\boldsymbol{\Sigma}_{xx}| - \ln |\boldsymbol{\Xi}_{zz}|.$$

Alternatively, we could use our current estimate of $\boldsymbol{\Sigma}_{z,x_i}$, namely, \mathbf{Z}_i . However, note that if the dimension of the missing data is larger than the dimension of the observed data for observation i then it will better to calculate $|\boldsymbol{\Sigma}_{xx}|$ directly.

3.7 Model Selection

The Bayesian information criterion (BIC; Schwarz, 1978) is used to select the number of components G and the number of latent factors q . For a model with parameters $\boldsymbol{\theta}$, $\text{BIC} = 2l(\mathbf{x}, \hat{\boldsymbol{\theta}}) - m \log n$, where $l(\mathbf{x}, \hat{\boldsymbol{\theta}})$ is the maximized log-likelihood, $\hat{\boldsymbol{\theta}}$ is the maximum likelihood estimate of $\boldsymbol{\theta}$, m is the number of free parameters in the model, and n is the number of observations. The use of the BIC in mixture model selection was originally (Dasgupta and Raftery, 1998) based on an approximation to Bayes factors (Kass and Raftery, 1995). The effectiveness of the BIC for choosing the number of factors in a factor analysis model has been established by Lopes and West (2004).

4 Analysis of the White Bread Data

4.1 The White Bread Data

A total of $n = 369$ consumers tasted six out of 12 white breads in a BIB design. Taste was evaluated on the hedonic scale, so values in $\{1, 2, \dots, 9\}$ are assigned to each tasted bread. For illustration, the first few rows of the data are shown in Table 1, where the bread brands are denoted A, B, \dots , L. We fitted our mixture of factor analyzers model, with common factors, to these data using the PEM algorithm introduced herein. These models were fitted for $G = 1, \dots, 6$ and $q = 1, \dots, 3$, using multiple restarts.

4.2 Results

The results (Table 2) show that the BIC selected a model with $G = 3$ components and $q = 2$ factors. Note that we also ran standard EM algorithms on these data and can confirm that they converged to the same results as our PEM algorithms. A plot of the two latent factors (Figure 1) shows the three components in the latent space. Because the classifications are based on maximum *a posteriori* (MAP) probabilities, it is

Table 1: The first six rows of the white bread data, where each consumer evaluates six breads using the hedonic scale.

Consumer	A	B	C	D	E	F	G	H	I	J	K	L
1	9		8	6				9			4	8
2	3		8		7		8	7	8			
3		8	6	7					6	9	7	
4			5	4		6		4	3	6		
5			7	7			8	7	6		8	
6				8			3	4	8		7	7

Table 2: BIC values from our analysis of the white bread data, for $G = 1, \dots, 6$ components and $q = 1, \dots, 3$ latent factors.

G	Number of Latent Factors		
	1	2	3
1	5273.9	5318.0	5369.9
2	5176.1	5136.0	5193.1
3	5148.1	5125.5	5244.1
4	5182.2	5171.5	5285.0
5	5223.1	5288.1	5341.7
6	5374.1	5439.1	5492.7

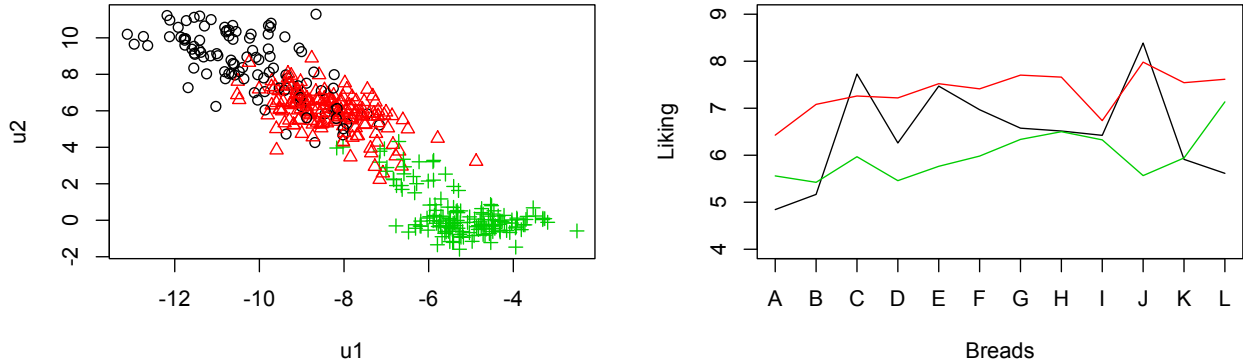


Figure 1: Plot of the two latent factors for the selected model, coloured by component (left), and a plot of the average liking scores for each of the breads separated by component (right).

straightforward to provide the client with probabilities rather than hard group memberships; this might be particularly desirable for consumers near the cluster boundaries.

In the same figure, there is also a plot of the mean liking scores for each bread for each of the three components. The red and green components seem to represent higher and lower scorers, respectively, with consumers within the black component exhibiting more variability in liking.

Some interesting points emerge from inspection of the results. Notably, bread J emerges as polarizing: it is strongly liked in the red and black groups and disliked in the green group. Interestingly, bread J is

the only ciabatta-style bread in the study and so it makes sense that its sensory properties will result in a relatively extreme liking response. This liking contrast is useful in differentiating groups of consumers because the objective of this research is to understand the sensory-based choice behaviour of consumers to define an optimum product for each liking cluster. Bread I is also interesting, in that it is the one bread for which consumers in all three groups seem to converge to the similar liking scores. Bread I is the sweetest, most flavourful bread in the study; it is also firm, dense, moist, and chewy. This is an unusual combination of characteristics and one would expect it to stand out. The fact that it stood out by not differentiating consumers in this study is itself interesting in the process of trying to understand the sensory-based choice behaviour of consumers.

4.3 Comparing PEM and EM

The analysis of the bread data was repeated using a standard EM algorithm for parameter estimation. The results were the same, as we would expect. Figure 2 illustrates the progression of the EM and PEM algorithms with $G = 1$ and $G = 2$ components, respectively, and $q = 2$ latent factors. As expected, both algorithms converge to the same solution in an almost identical fashion.

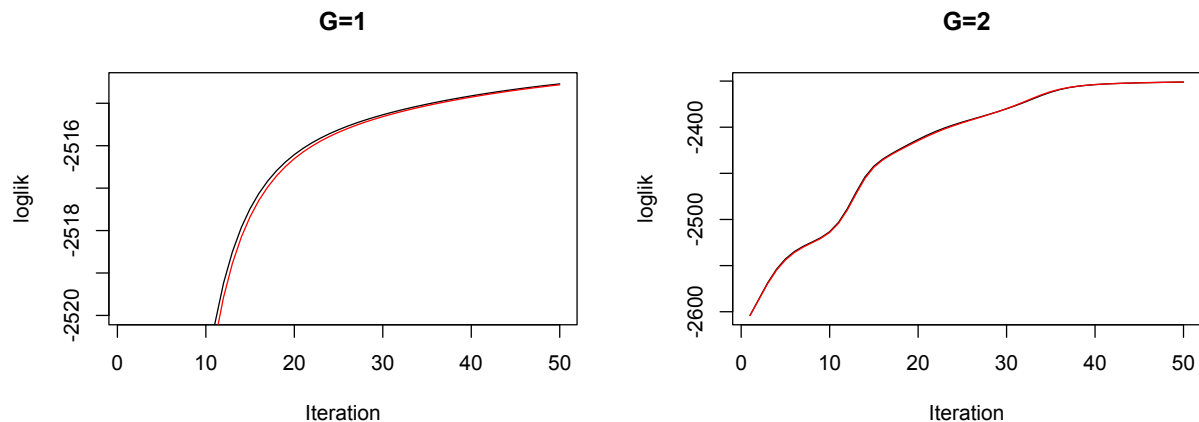


Figure 2: Plot of the log-likelihood for $G = 1$ and $G = 2$ for PEM (red line) and EM (black line) algorithms.

5 Discussion

We developed an approach for clustering incomplete BIB data from consumer tasting of 12 different commercial white breads. Our clustering approach is based on a parsimonious mixture of factor analyzers model, where the factor loading matrices are constrained to be equal across groups. The problem of missing data is handled along with parameter estimation within an partial EM algorithm framework. Rather than simple imputation, this PEM algorithm approach effectively imputes missing data at each iteration based on current component membership probabilities; this is a natural approach as missing values are filled in based on complete values in observations that are in some sense similar (i.e., in the same component). Our PEM algorithm is much more computationally efficient than the standard EM algorithm for this, and any such, missing data problem. The PEM is shown to retain the monotonicity property and, thus, retains the same convergence properties as the EM algorithm. Three benefits are achieved through this approach: the quality of data that are collected prior to fatigue is improved; the method of substituting missing data reflects

the sensory preferences of each consumer, which permits robust cluster assignment; and the collection of incomplete-block data reduces the cost, time, and materials required for this type of study.

We introduce a new variation of the EM algorithm called the PEM algorithm. The many varieties of the EM mainly focus on the M-step: the expectation-conditional maximization (ECM) algorithm (Meng and Rubin, 1993), the ECM either (ECME) algorithm (Liu and Rubin, 1994), the alternating ECM (AECM) algorithm, and others. Few examine different ways of partially updating the E-step. Neal and Hinton (1998) give several possible methods to update the missing sufficient statistics. All of the methods suggest something along the lines of fully updating a partial set of the missing sufficient statistics. When the E-step is intractable, Wei and Tanner (1990) suggest approximating the E-step by simulating m observations from the conditional distribution of the missing data given the observed data. This version of EM algorithm is called Monte Carlo EM (MCEM). Prior to MCEM, Celeux and Diebolt (1985) suggested using stochastic EM (SEM), which is the same as MCEM with $m = 1$. Other variations on approximating the E-step have been introduced, such as MCEM using rejection sampling, importance sampling, and Markov chain Monte Carlo. The PEM algorithm presented here is similar to ECM in which we have ‘conditional’ E-steps instead of conditional M-steps. These conditional E-steps are computationally cheaper than using a complete or full E-step.

Acknowledgements

This work was supported by a grant-in-aid from Compusense Inc. and by a Collaborative Research and Development grant from the Natural Sciences and Engineering Research Council of Canada.

A Some Mathematical Details

A.1 Schur Complement Relation in Quadratic Form

Suppose we have a positive-definite symmetric matrix \mathbf{S} and a vector \mathbf{y} with decompositions

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix} \quad \text{and} \quad \mathbf{S} = \begin{bmatrix} \mathbf{S}_{11} & \mathbf{S}_{12} \\ \mathbf{S}_{21} & \mathbf{S}_{22} \end{bmatrix},$$

then

$$\mathbf{y}'\mathbf{S}^{-1}\mathbf{y} = \mathbf{y}_1'\mathbf{S}_{11}^{-1}\mathbf{y}_1 + (\mathbf{y}_2 - \mathbf{S}_{21}\mathbf{S}_{11}^{-1}\mathbf{y}_1)'\mathbf{S}_{22.1}^{-1}(\mathbf{y}_2 - \mathbf{S}_{21}\mathbf{S}_{11}^{-1}\mathbf{y}_1),$$

where $\mathbf{S}_{22.1} = \mathbf{S}_{22} - \mathbf{S}_{21}\mathbf{S}_{11}^{-1}\mathbf{S}_{12}$.

A.2 A Matrix Minimization Problem

Suppose we have a positive-definite symmetric matrix \mathbf{S} with decomposition

$$\mathbf{S} = \begin{bmatrix} \mathbf{S}_{11} & \mathbf{S}_{12} \\ \mathbf{S}_{21} & \mathbf{S}_{22} \end{bmatrix}.$$

We then have the following property for a function γ ,

$$h(\boldsymbol{\Theta}_{11}) = \text{tr} \{ (\mathbf{S}_{11} - \boldsymbol{\Theta}_{11}, \mathbf{S}_{12}) \mathbf{S}^{-1} (\mathbf{S}_{11} - \boldsymbol{\Theta}_{11}, \mathbf{S}_{12}) \} \geq \text{tr} \{ \mathbf{S}_{22}^{-1} \mathbf{S}_{12} \mathbf{S}_{12}' \}. \quad (17)$$

Equality holds when $\boldsymbol{\Theta}_{11} = \mathbf{S}_{11} - \mathbf{S}_{12}\mathbf{S}_{22}^{-1}\mathbf{S}_{21}$. Therefore, $h(\boldsymbol{\Theta}_{11})$ is minimized by the Schur complement $\boldsymbol{\Theta}_{11} = \mathbf{S}_{11} - \mathbf{S}_{12}\mathbf{S}_{22}^{-1}\mathbf{S}_{21}$. Now, if we define

$$\boldsymbol{\Theta} = \begin{bmatrix} \boldsymbol{\Theta}_{11} & 0 \\ 0 & 0 \end{bmatrix}, \quad (18)$$

then

$$h(\Theta_{11}) = \text{tr} \{ (\mathbf{S} - \Theta) \mathbf{S}^{-1} (\mathbf{S} - \Theta) \} = \gamma(\Theta_{11}) + \text{tr} \{ \mathbf{S}_{22}^{-1} \mathbf{S}_{22} \mathbf{S}_{22} \}. \quad (19)$$

Because the right-hand term does not depend on Θ_{11} , $h(\Theta_{11})$ has the same minimum as $g(\Theta_{11})$; i.e., the Schur complement $\Theta_{11} = \mathbf{S}_{11} - \mathbf{S}_{12} \mathbf{S}_{22}^{-1} \mathbf{S}_{21}$. Therefore, a minimization algorithm based on the function h is equivalent to minimizing γ . We minimize h using a conditional minimization algorithm (by column/row) based on Equation (17).

References

- Bartlett, M. (1953). Factor analysis in psychology as a statistician sees it. In *Uppsala Symposium on Psychological Factor Analysis*, Number 3 in Nordisk Psykologi's Monograph Series, Uppsala, Sweden, pp. 23–34. Almqvist and Wiksell Uppsala.
- Celeux, G. and J. Diebolt (1985). The SEM algorithm: A probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. *Computational Statistics Quarterly* 2, 73–82.
- Celeux, G. and G. Govaert (1995). Gaussian parsimonious clustering models. *Pattern Recognition* 28, 781–793.
- Dasgupta, A. and A. E. Raftery (1998). Detecting features in spatial point processes with clutter via model-based clustering. *Journal of the American Statistical Association* 93, 294–302.
- Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B* 39, 1–38.
- Findlay, C. J. (2008). Consumer segmentation of BIB liking data of 12 cabernet sauvignon wines: A case study. Presented at the 9th Sensometrics Meeting, July 20–23, St. Catharines, Canada.
- Fraley, C. and A. E. Raftery (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association* 97, 611–631.
- Ghahramani, Z. and G. E. Hinton (1997). The EM algorithm for factor analyzers. Technical Report CRG-TR-96-1, University of Toronto.
- Hunter, D. L. and K. Lange (2000). Rejoinder to discussion of “Optimization transfer using surrogate objective functions”. *Journal of Computational and Graphical Statistics* 9, 52–59.
- Hunter, D. L. and K. Lange (2004). A tutorial on MM algorithms. *The American Statistician* 58(1), 30–37.
- Jaeger, S. R., K. L. Rossiter, W. V. Wismer, and F. R. Harker (2003). Consumer-driven product development in the kiwifruit industry. *Food Quality and Preference* 14(3), 187–198.
- Kass, R. E. and A. E. Raftery (1995). Bayes factors. *Journal of the American Statistical Association* 90, 773–795.
- Lawless, H. T. and H. Heymann (2010). *Sensory Evaluation of Food: Principles and Practices*. New York: Springer.
- Liu, C. and D. B. Rubin (1994). The ECME algorithm: A simple extension of EM and ECM with faster monotone convergence. *Biometrika* 81, 19–39.
- Lopes, H. F. and M. West (2004). Bayesian model assessment in factor analysis. *Statistica Sinica* 14, 41–67.

- Magnus, J. R. and H. Neudecker (1998). *Matrix Differential Calculus with Applications in Statistics and Econometrics*. New York: John Wiley & Sons.
- McLachlan, G. J. and K. E. Basford (1988). *Mixture Models: Inference and Applications to Clustering*. New York: Marcel Dekker Inc.
- McLachlan, G. J. and D. Peel (2000). Mixtures of factor analyzers. In *Proceedings of the Seventh International Conference on Machine Learning*, San Francisco, pp. 599–606. Morgan Kaufmann.
- McNicholas, P. D. and T. B. Murphy (2008). Parsimonious Gaussian mixture models. *Statistics and Computing* 18, 285–296.
- Meng, X. L. and D. B. Rubin (1993). Maximum likelihood estimation via the ECM algorithm: a general framework. *Biometrika* 80, 267–278.
- Neal, R. M. and G. E. Hinton (1998). A view of the EM algorithm that justifies incremental, sparse, and other variants. In M. I. Jordan (Ed.), *Learning in Graphical Models*, pp. 335–368. Dordrecht: Kluwer Academic Publishers.
- Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics* 6, 461–464.
- Spearman, C. (1904). The proof and measurement of association between two things. *The American Journal of Psychology* 15(1), 72–101.
- Wei, G. and M. A. Tanner (1990). A Monte Carlo implementation of the EM algorithm and the poor man’s data augmentation algorithms. *Journal of the American Statistical Association* 85(411), 699–704.
- Wolfe, J. H. (1963). Object cluster analysis of social areas. Master’s thesis, University of California, Berkeley.